面向 XML 网关/防火墙产品的 RSA 安全实施指南

最后修订：2004 年 9 月 29 日

1. 合作伙伴信息

合作伙伴名称	IBM
网站	http://www.ibm.com/software/cn/websphere/datapower/
产品名称	DataPower XS40 XML Security Gateway™
版本 & 平台	3.1.2
产品说明	DataPower XS40 XML Security Gateway™ 是一款定制的 1 U 机架式网络设备，提供了全面的 XML 安全功能集合，包括：XML 加密、XML/SOAP 防火墙过滤、XML 数字签名、XML 架构验证、SSL、XML 访问控制。
产品分类	Web 服务

2. 联系人信息

	销售联系人	支持联系人
	李宗灿	崔鹏
电子邮箱	johnlee@cn.ibm.com	cuipeng@cn.ibm.com
电话	(86-10)63618826	(86-10)63612176
网站	http://www.ibm.com/software/cn/websphere/datapower/	http://www.ibm.com/software/cn/websphere/datapower/

3. 解决方案摘要

特性	详细资料
面向 **XML** 网关**/**防火墙应用程序 担当 **SAML** 断言方的 **RSA FIM**	是
XML 网关**/**防火墙产品可以 将 **SAML** 断言附加到现有 **SOAP** 信息中	是
两种产品同时支持的 **SAML** 版本	**1.1**
受支持的 **Web SSO** 文件	**BAP**

4. 整合概述

DataPower XS40 Security Gateway 通过两种方式实现了 SAML 与 RSA Federated Identity Manager (FIM) 的互操作性。

首先，DataPower XS40 Gateway 能够向 RSA FIM 发出 SAML 属性查询，以获得已验证身份的属性信息。这些属性具有“Admin”或“Title = Manager”这样的形式。基于这些属性，XS40 能够规定已验证的身份可获准访问哪些 URL。

例如，只有管理人员被允许访问“ListAddress”URL。一条未提供来自管理员的有效凭据的 SOAP 请求将被拒绝访问 ListAddress。通过这种方式，同时使用 RSA FIM 和 DataPower，能够细粒度地控制对有价值的 XML Web 服务的访问。

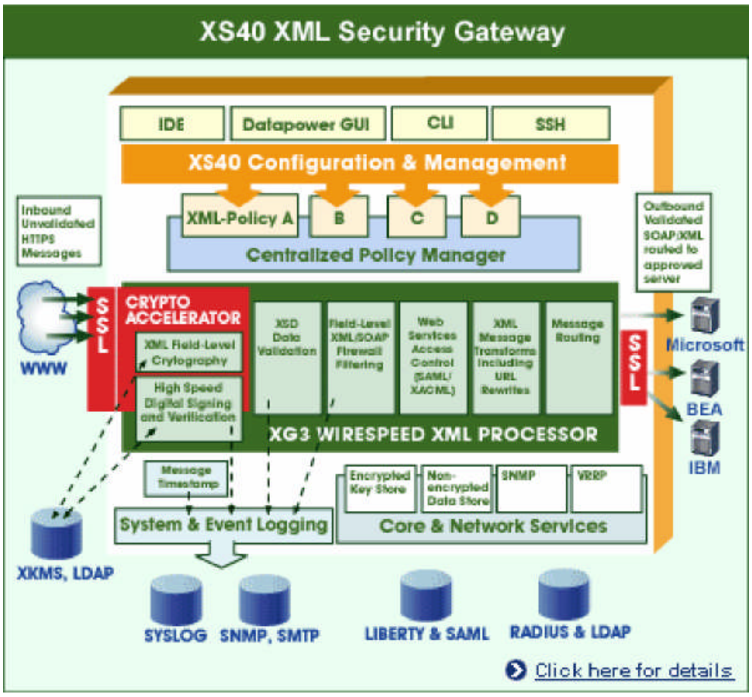
其次，DataPower XS40 使用 SAML Browser/Artifact Profile，允许用户在本地 Web 站点（例如门户）连接主机，并将身份验证信息发送至远程 Web 站点（例如来自合作伙伴 Web 站点的联合应用程序）。这为用户提供了 Web 单点登录（SSO）体验，因为用户无须再次接受远程 Web 站点验证。XS40 能够用于保护远程 Web 站点，并从 RSA FIM 获得 SAML 身份验证信息。

在 SAML Browser/Artifact Profile 中，用户首先在本地 Web 站点验证。然后当他单击远程 Web 站点链接时，RSA FIM 将重定向用户至 XS40，并传递标识已验证身份的浏览器工件。然后 XS40 能够向 RSA FIM 发布 SAML 请求，并检索包含已验证身份的 SAML 断言。然后 XS40 能够使用此 SAML 断言为远程 Web 站点用户设置 cookie。

5. 产品需求

硬件需求

组件名称：	
DataPower Firmware	Release 3.1.2 （或更新版本）



6. 产品配置

DataPower XS40 防火墙能够配置为发送 SAML 属性查询，或通过 SAML Browser/Artifact Profile 从 RSA Federated Identity Manager (FIM)中检索 SAML 断言。我们将在 6.2 和 6.3 节中分别分析如何设置两种类型的 SAML 支持。

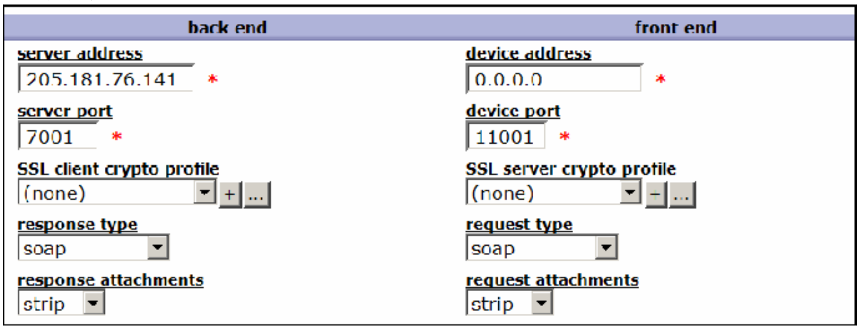
在配置 DataPower XS40 之前，您应该确认您的 RSA FIM 服务器被配置为可根据希望检验的身份或主题，从 RSA ClearTrust 中检索适当属性。欲获取更多信息，请参看《*RSA Federated Identity Manager 2.5 规划和安装指南》*。

6.1 HTTP Basic Authentication 的助手防火墙

在这两个例子中，与 RSA Federated Identity Manager (FIM)服务器的通信均通过 HTTP BASIC Authentication 完成的。为实现这一目的，我们首先需要安装可获取适当参数（用户名和密码）并创建 HTTP BASIC Authentication 报头的助手防火墙。您还能够利用 SSL 与 RSA FIM 服务器通信。欲了解更多关于此方式的详情，参看下面标题为启用对 RSA Federated Identity Manager (FIM)的 SSL 通信的章节。

在控制面板上，单击 **New XML Firewall**，然后单击 **Advanced**，再单击 **Next**。

在显示的防火墙界面上，输入防火墙名称。后端服务器应该设置为 RSA Federated Identity Manager (FIM)服务器的 IP 地址和端口。前端服务器可以使用默认地址和端口。记住端口数字以便将来使用。



对于防火墙策略，使用匹配规则以匹配所有 URL，并使用转换来添加 HTTP BASIC Authentication 报头。欲获取此转换所需 XML 配置文件，请联系 DataPower。

保存 防火墙。

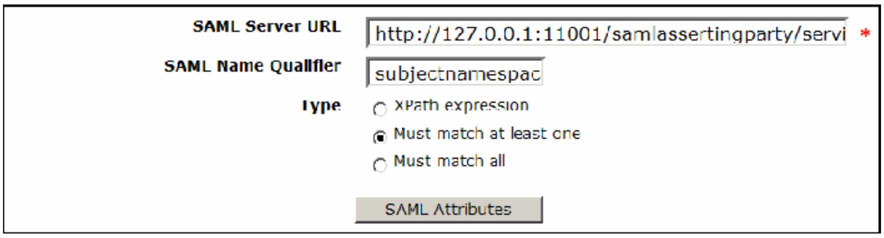
关于创建防火墙的更多信息，请参考 *XS40 XML Security Gateway WebGUI 入门*。

6.2 SAML 属性查询防火墙

在创建上述 HTTP Basic Authentication 助手防火墙之后，我们能够创建一个用于身份验证、授权、审计（AAA）的防火墙，此防火墙使用对 RSA Federated Identity Manager (FIM)服务器的 SAML 查询来进行授权。

1. 在控制面板上，单击 **New XML Firewall**，然后单击 **AAA**，在单击 **Next**。
2. 为其命名，单击 **Next**。
3. 选择 **Static-backend** 并单击 **Next**。

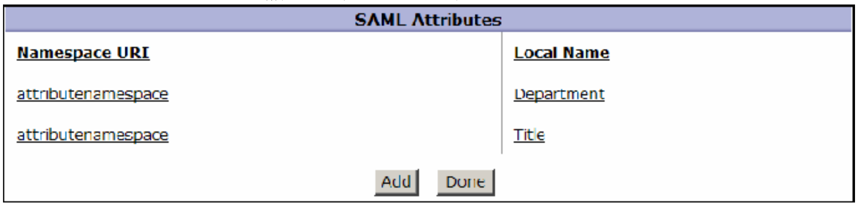
4. 输入您要保护的后端服务器的相关信息，单击 **Next**。
5. 您可以使用默认地址 0.0.0.0 作为服务器地址，并为防火墙设置您想公开给外界的端口，再单击 **Next**。
6. 现在显示了一个现有 AAA 策略的列表。单击**+**以创建新策略。
7. 在弹出框中为 AAA 策略命名。
8. 根据您的用户方案选择提取身份和验证的方式。
9. 在授权步骤中，选择 Method、**Generate a SAML attribute query**。然后您将看到一个供选择的字段列表。



10. 为 SAML Server URL 输入 http://127.0.0.1:<助手防火墙端口>/< RSA FIM SAML 请求的 URL 的剩余部分>

（例如，http://1 27.0.0.1:11 001/samlassertingparty/services/SamlRequest?username=relying_party&password=password）

11. 输入适当的 SAML 名称限定符，无论输入的是是部分还是全部属性都应匹配。
12. 单击 **SAML Attributes**，输入您关心的属性。



开箱即用的 XS40 能够检查这些属性是否存在。为检查这些属性的值是否匹配预期值，您可以使用自定义配置文件。使用 XS40 DataPower 用户支持来查看如何获取针对此目标的自定义配置。

13. 单击 **Next**，然后单击 **Commit**。
14. 现在回到主界面，您应该能看到界面上显示了刚刚创建的 AAA 策略名称。
15. 单击 **Next**，然后 **Commit**。

注意：若使用 SSL ，请跳过 6.1 节，并使用主 XS40 防火墙中的 SSL 服务器加密文件字段。更多信息请参看 XS40 文档。在步骤 10 中，您应该直接使用 RSA FIM 服务器的 IP 地址，而非 127.0.0.1。

6.2.1 属性查询示例日志

下面是您应该查看的成功授权结果范例，假定存在可回送 XML 消息的受保护的后端服务器。下面使用的端口数字与上述步骤 9 相同：

```
C:\TEMP>curl -u uid=dp1:pass1234 -d @message.xml http://10.10.35.53:2048/
<?xml version="1.0" encoding="UTF-8"?>
<message xsi:schemaLocation="http://www.example.com message.xml" xmlns="http://w
ww.example.com" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">   <to>Al
ce</to>   <from>Bob</from>   <subject>Important</subject>   <body>This is an im
ortant message</body></message>
```

如果用户姓名/密码对不正确，应该显示一个错误（被策略拒绝）：

```
C:\TEMP>curl -u uid=dp1:pass1233 -d @message.xml http://10.10.35.53:2048/
<?xml version="1.0" encoding="UTF-8"?>
<message xsi:schemaLocation="http://www.example.com message.xml" xmlns="http://w
ww.example.com" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">   <to>Al
ce</to>   <from>Bob</from>   <subject>Important</subject>   <body>This is an im
ortant message</body></message>HTTP/1.1 500 Internal Server Error
Connection: close
Content-Type: text/xml

<?xml version='1.0' ?>
<env:Envelope xmlns:env='http://schemas.xmlsoap.org/soap/envelope/'>
<env:Body>
<env:Fault>
<faultcode>env:Client</faultcode>
<faultstring>Rejected by policy. (from client)</faultstring>
</env:Fault>
</env:Body>
</env:Envelope>
```

下面是 XS40 日志的代码片段，显示了一次成功的 SAML 属性查询。在本例中，从底部开始，205.181.76.141 生成对 RSA FIM 服务器的请求，助手防火墙解析该输出，然后主防火墙解析该输出，最后发布授权（Authorizing with "saml-attr"）：

```
aaa          debug XMLFirewallService (INPUT): Policy(test-SAML-attribute-
                   query): Authorizing with "saml-attr"
xmlparse     debug Parsing document
                    'http: //127.0.0.1: 11001/samlassertingparty/services/Sam
                   lRequest? username=relying_party&password=password'
Xmlfirewall  debug ...
multistep    info xmlfirewall (add-basic-auth) : response add-basic-
                   auth _Rule _0 #1 xform: 'Transforming INPUT with
                   local:///add-basic-auth.xsl results stored in OUTPUT'
                   completed ok.
xslt         debug ...
xslt         debug ...
multistep    debug ...
xmlparse     debug Parsing document
                    'http: //205 .181.76.141: 7001/samlassertingparty/services
                   /SamlRequest? username=relying_party&password=password'
schema       debug ..
xmlfirewall debug xmlfirewall (add-basic-auth) : HTTP response code sent
                   to client: 200 OK (url
```

```
                    http: //205.181.76.141: 7001/samlassertingparty/services/
                    SamlRequest? username=relying_party&password=password)
schema      debug ..
```

6.3 SAML Browser Artifact Profile 防火墙

为了支持 SAML Browser/Artifact Profile (BAP)，DataPower XS40 在保护远程 Web 站点方面充当了 Artifact Consumer 角色，如下图所示，下图出自针对 SAML V1 .1 的 OASIS *技术概览*文档。RSA Federated Identity Manager (FIM)可通过站点间传输和响应程序服务履行保护本地 Web 站点的职责。

首先，确认您已为站点间传输和响应程序服务正确配置了 RSA FIM ，并将 XS40 注册为受信任的依赖方。更多信息请参看《*RSA Federated Identity Manager 2.0 规划和安装指南*》。

然后我们将在 XS40 上安装防火墙，接受图中步骤#7 的工件，使用 SAML 请求产生#8 调用，以进行身份验证，解析步骤#9 的响应，并执行步骤#10 的重定向。

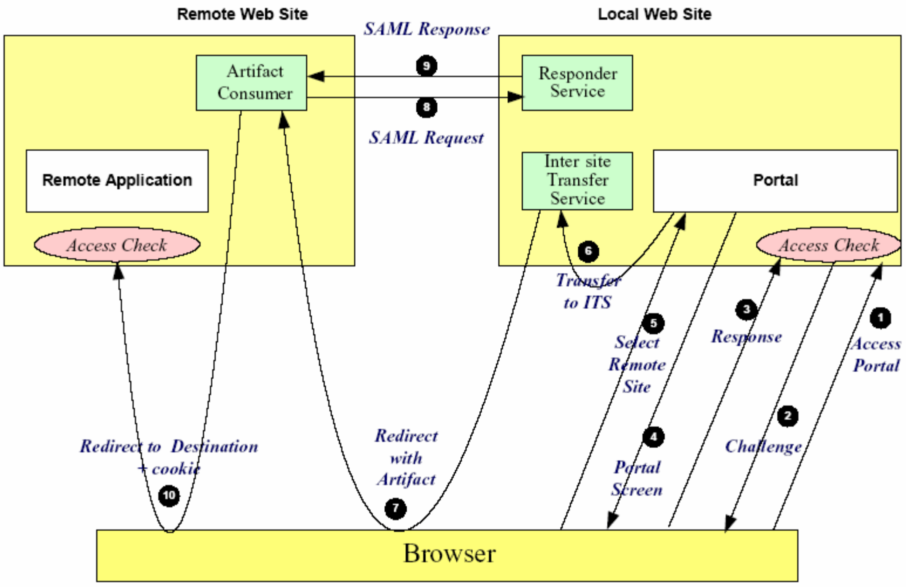


Figure 9: Browser/Artifact Profile - Detailed Processing

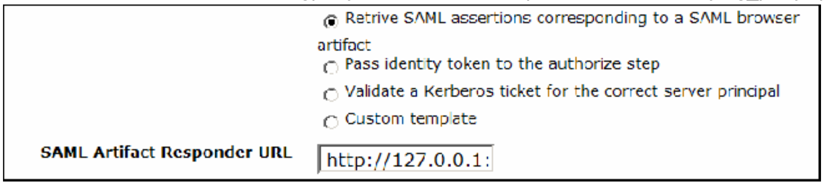
版权所有 © OASIS Open 2004. *保留所有权利。*
本文档及其译文可被复制并提供给他人，衍生著作（评论、说明或实施帮助）可以无限制地以节选或全文形式进行筹备、复制、出版和分发，但上述版权声明和本段落必须包含在副本和衍生著作中。但是，不允许对本文本身进行任何形式的修改，例如移除版权声明或对 OASIS 的引用，除非出于在遵循 OASIS Intellectual Property Rights 文档中定义的版权程序的情况下编写 OASIS 说明的目的，或者处于将其翻译为英语之外的语言的需要。

要使用 HTTP Basic Authentication，首先确认您已经根据上述 6.1 节为 HTTP 身份验证创建了助手防火墙。然后我们将使用以下步骤创建 AAA 防火墙。

1. 打开控制面板，单击 **New XML Firewall**，然后单击 **AAA**，再单击 **Next**。
2. 为其命名并单击 **Next**。
3. 选择 **Static-backend** 并单击 **Next**。
4. 输入你要保护的后端服务器的相关信息，单击 **Next**。
5. 您可以使用默认地址 0.0.0.0 作为服务器地址，并为防火墙设置您想向外界公开的端口，再单击 **Next**。
6. 现在显示了一个现有 AAA 策略大列表。单击**+**以创建新策略。
7. 在弹出框中为 AAA 策略命名。
8. 在身份方法中选择 **SAML Artifact**。
9. 选择您希望的映射凭据方式，如果不确定则使用默认。
10. 在身份验证方法中选择 **Retrieve SAML assertions corresponding to a SAML browser artifact**，并输入 SAML Artifact Responder URL：

    http://127.0.0.1:<助手防火墙端口>/< RSA FIM SAML 请求的 URL 的剩余部分>

（例如，http://1 27.0.0.1:11 001/samlassertingparty/services/SamlRequest?username=relying_party&password=password）

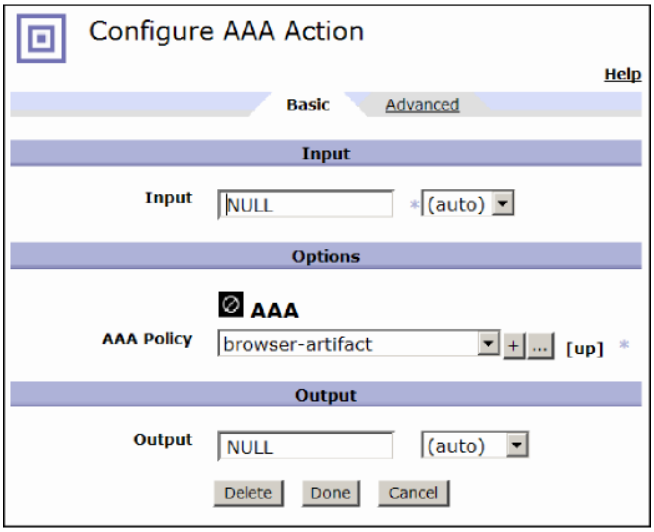


11. 在资源步骤中，选择 **URL sent by client** 和 **Custom** 以将资源映射为目标 URL。联系 DataPower 以获取您能用于此处的配置文件。然后单击 **Next**。
12. 在授权步骤中，选择 **Allow any authenticated client**。
13. 单击 **Next**，然后单击 **Commit**。
14. 现在回到主界面，您应该能看到刚刚创建的 AAA 策略名称。
15. 单击 **Next**，然后单击 **Commit**。

注意： 若使用 SSL 与 RSA FIM 服务器通信，请使用主 XS40 防火墙中的 SSL 服务器加密文件字段。更多信息请参看 6.3.3 节和 XS40 文档。

AAA 防火墙已经创建了，但我们需要进行一些修改，使其能使用 SAML 浏览器工件查询。首先，我们将修改防火墙规则。

16. 在控制面板上，单击 **Edit XML Firewall**，然后单击您刚创建的防火墙名称。
17. 对于 Firewall Policy 字段，单击 **...** 图标。
18. 在弹出框中，选择 **Client to Server** 单选按钮。
19. 双击细线上的 图标，打开 AAA 步骤配置菜单。



20. 对于 Input 和 Output 字段，将值均改为 **NULL**，然后单击 **Done**。

21. 双击细线上的图标，打开 Transform 配置菜单，再单击 **Delete**。

22. 现在您的细线上应该仅剩两个图标了：图标和图标。

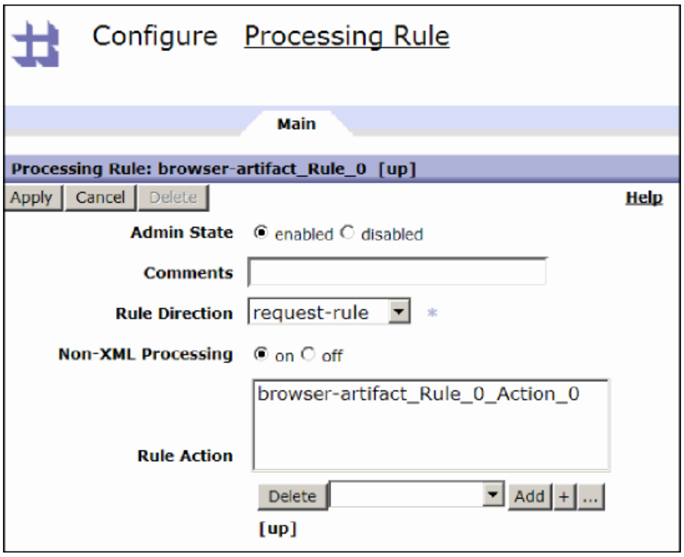
    然后单击 **Apply** 应用规则，单击 **Close** 关闭弹出框。

23. 现在回到主界面，在 Front End Request Type 中选择 **XML** 而不是默认 SOAP 类型。

24. 单击 **Next**，再单击 **Next**，随后单击 **Commit**。

    接下来，我们需要进行另外的局部编辑：

25. 在 Objects 菜单中，单击 **Processing Policy**，并选择您已创建的防火墙名称。
26. 然后单击 **Policy Maps** 标签，再单击 **Edit**。
27. 在弹出框中选择规则，选择 **...** 图标，您应该看到如下菜单。



28. 在 Non-XML Processing 栏中，选择 **On**，然后单击 **Apply**，再单击 **Close** 关闭菜单。

防火墙现在已正确设置，可支持 SAML Browser/Artifact Profile。

6.3.1 浏览器工件结果

现在转到受 RSA FIM 保护的本地 Web 站点，在本地进行验证，单击链接转至被 XS40 保护的远程 Web 站点上的页面，（SAML 浏览器工件图中的步骤 #1-5）。RSA FIM 将重定向浏览器至您已创建的 XS40 防火墙。然后 XS40 将向 RSA FIM 生成 SAML 请求，以获得身份验证断言，并将您重定向至远程 Web 站点中的页面。

下面是您应该能在日志中看到的内容，按时间倒序排列。从底部向上阅读，您将看到提取浏览器工件的条目，在 205.181.76.141 上对 RSA FIM 服务器生成请求，在 127.0.0.1:11001 上使用助手防火墙。接着收到 SAML 访问并缓存成功的身份验证。

```
aaa      debug 8775 ...
aaa      debug 8775 XMLFirewallService (NULL): Policy(browser-artifact):
                    Cached AU entry
xmlpars  debug      …
e
xmlf ire debug      xmlfirewall (add-basic-auth) : HTTP response code sent to
wall                client: 200 OK (url
                    http: //205.181.76.141: 7001/samlassertingparty/services/S
                    amlRequest? username=relying_party&password=password)
xmlf ire info       …
wall
xmlf ire debug      xmlfirewall (add-basic-auth) : Server side connection
wall                made for POST
                    http: //205 .181.76.141: 7001/samlassertingparty/services/S
                    amlRequest? username=relying_party&password=password to
                    205.181.76.141:7001. Obtaining Response.
network  debug      …
xmlf ire debug      xmlfirewall (add-basic-auth) : New transaction (conn
wall                use=1) : POST
                    http: //127 .0.0.1: 11001/samlassertingparty/services/SamlR
                    equest? username=relying_party&password=password from
                    127.0.0.1
network  debug      …
aaa      debug 8775 XMLFirewallService (NULL): Policy(browser-artifact):
                    Authenticating with "saml-artifact"
aaa      debug 8775 XMLFirewallService (NULL): Policy(browser-artifact) : AU
                    cache check with key="policyname<?xml version="1.0"
                    encoding="UTF-8"?> <identity><entry type="saml-
                    artifact"><artifact>AAGwSESsHTJlhtmlZSYT6rRW5tH1R4MwLFQK
                    SXQXjXM+cNZcomMVHTCc</artifact></entry></identity><? xml
                    version="1 .0" encoding="UTF-8"?> <au-ancillary-
                    info/> saml-artifact"
aaa      debug 8775 ...
aaa      debug 8775 XMLFirewallService (NULL): Policy(browser-artifact):
                    SAML artifact:
                    AAGwSESsHTJlhtmlZSYT6rRW5tH1R4MwLFQKSXQXj XM+cNZcomMVHTCc
xmlf ire debug      …
wall
xmlf ire debug      xmlfirewall (browser-artifact): New transaction (conn
wall                use=1) : GET
```

http://burnin.datapower.com:7001/SAMLArtifact?SAMLart=AAGwSESsHTJlhtmlZSYT6rRW5tH1R4MwLFQKSXQXj XM%2BcNZcomMVHTCc&TARGET=http%3A%2F%2Fburnin . datapower. com%3A9000%2Ftarget from 10.10.1.178

6.3.2 测试来自 RSA Federated Identity Manager (FIM)的已签名响应

对来自 RSA Federated Identity Manager (FIM)的 SAML 响应进行签名，这对确保响应的真实性很有用。要测试来自 RSA FIM 的已签名响应，您需要

o 在 RSA FIM 一侧启用签名响应，
o 从 XS40 上的 RSA FIM 安装 X.509 签名证书，
o 更改助手防火墙以验证 RSA FIM 所返回的响应的签名。

让我们练习一下这些步骤。首先，在 RSA FIM Administrative UI 上启用签名。编辑受信任的依赖方数字签名栏，从而为响应签名。

注意： 目前不支持断言的签名。

1. 对于 Responses，勾选 **Sign responses sent to this trusted relying party** 。
2. 对于 Response Signature Keystore，选择合适的密钥。
3. 对于 Key information 字段，选择 **The verifying certificate** 。
4. 单击 **Save**。

现在将用于对响应进行签名的 X.509 证书复制到 XS40。参看附录 B 中的 *XS40 XML Security Gateway WebGUI* 入门，以获得更多关于创建加密证书和加密验证凭据对象的信息。

5. 首先使用 X.509 证书创建加密证书对象。
6. 然后使用加密证书对象创建加密验证凭据对象。

现在编辑 HTTP Basic Authentication 助手防火墙，以添加对来自 RSA FIM 服务器的合法签名的校验。

7. 从控制面板中选择 **Edit XML Firewall**，单击助手防火墙名称。
8. 在 Firewall Policy 下，单击 **...**。
9. 在弹出框中，对于 Rule Actions，单击 **New**，然后单击 **OK**。
10. 双击细线上的图标，使用 * 选择一条匹配所有 URL 的规则。
11. 拖动到细线上，并双击它。
12. 在弹出框中，对于 Validation Credential，选择您在步骤 6 中创建的加密有效凭据对象。然后单击 **Done**。
13. 选择单选按钮 **Server to Client**，然后单击 **Apply**。

14. 现在单击 **Close** 关闭弹出框。

现在重新运行测试，进入本地 Web 站点并单击远程站点链接。您应该在日志中看到新的签名验证条目，如下所示。

```
aaa          debug 9802 XMLFirewallService (NULL): Policy(browser-artifact) : Cached AU
                          entry
multistep info 9803 ...
crypto      info 9803 xmlfirewall (add-basic-auth) : Signature verification done
xmlfilter warn 9803 xmlfirewall (add-basic-auth) : Accept set
xmlfilter warn 9803 xmlfirewall (add-basic-auth) : Accept set
schema      debug      …
xmlfirewall debug      xmlfirewall (add-basic-auth) : HTTP response code sent to client:
                       200 OK (url
                       http://205.181.76.141: 7001/samlassertingparty/services/SamlReque
                       st? username=relying_party&password=password)
xmlfirewall info       …
```

作为测试，如果您现在转到 RSA FIM Admin UI 并关闭响应签名，那么当单击远程 Web 站点链接时，您将看到“Rejected by policy”错误。

6.3.3 启用对 RSA Federated Identity Manager (FIM) 的 SSL 通信

为确保数据机密性，您可能希望启用 SSL 来进行 XS40 与 RSA FIM 之间的通信。为实现此目的，您需要首先在 XS40 上安装 SSL 证书。

1. 首先使用 RSA FIM 的 X.509 证书创建加密证书对象。
2. 然后使用加密证书对象创建加密有效凭据对象。
3. 接着使用有效凭据对象创建加密配置文件。继而设置 HTTP Basic Authentication 助手防

火墙以使用 SSL。

4. 从控制面板中选择 **Edit XML Firewall**， 再单击助手防火墙名称。
5. 对于 Backend，更改服务器端口，为 RSA FIM 使用 SSL 端口。例如将端口 7001 改为 7002。
6. 同样对于 Backend，将 SSL 客户机加密配置文件选为您在步骤 3 中所创建的配置文件。
7. 单击 **Next**，再单击 **Next**，然后单击 **Commit**。

参看附录 B 中的 XS40 XML Security Gateway WebGUI 入门，以获取更多关于创建这些对象的信息。

现在如果重新运行测试，您应该看到 XS40 日志中有针对 SSL 的条目。

```
xmlfirewall debug xmlfirewall (add-basic-auth) : HTTP server side response code: 200
                  OK (url
                  http://205.181.76.141: 7002/samlassertingparty/services/SamlRequ est?
                  username=relying_party&pas sword=pas sword)
ssl          info  SSL: certificate for '/C=US/O=RSA/OU=PE -LAB/CN=pe141.pe-
                  lab.com/emailAddress=pe141@pe134.pe -lab.com' is valid
xmlfirewall debug xmlfirewall (add-basic-auth) : Server side connection made for POST
                  http://205.181.76.141: 7002/samlassertingparty/services/SamlRequest?
                  username=relying_party&password=password to 205.181.76.141:7002.
                  Obtaining Response.
```

7. XML 网关/防火墙产品的认证清单

测试日期：**2004 年 8 月 27 日**

产品	已测试版本
RSA Federated Identity Manager (FIM)	2.5
RSA ClearTrust	5.5.2
DataPower XS40 XML Security Gateway	Firmware Release 3.1.2

测试用例	结果	
注意：*下列测试用例假定 RSA Federated Identity Manager (FIM) 被配置为**断言方** (AP)，而合作伙伴产品被配置为**依赖方** (RP)。*		
SAML 依赖方 (RP)	**SAML 1.0**	**SAML 1.1**
RSA FIM **生成**有效身份验证断言，以响应合作伙伴产品的有效身份验证查询		N/A
合作伙伴产品根据对 FIM 的有效身份验证查询的要求，**使用**有效身份验证断言		N/A
RSA FIM **生成**有效属性断言，以有效响应合作伙伴产品的属性查询		P
合作伙伴产品根据对 FIM 的有效属性查询的要求，**使用**有效属性断言		P
RSA FIM **生成**有效断言，以有效响应合作伙伴产品的 Assertion IDReference 请求，		N/A
合作伙伴产品根据对 FIM 的 Assertion IDReference 请求的要求，**使用**有效断言		N/A
Web 浏览器 SSO 配置文件		
Browser/Artifact Profile (BAP)		
生成有效断言以响应 AssertionArtifact 请求		P
与 HTTP 消息 HTTP BASIC Authentication 中所发送工件相对应的有效断言请求		P
匿名 SSL		P
Mutual Auth SSL		N/A
用户无须再次输入凭据就能被应用程序所验证		P**
发送至合作伙伴产品（RP）并经过验证的有效已签名响应		P
发送至合作伙伴产品（RP）并经过验证的有效已签名断言		N/A
来自合作伙伴产品（RP）的已签名请求的成功验证		N/A
Web SSO 断言中属性语句的成功验证		N/A
Browser/POST Profile (BPP)		
在有效 HTTP POST 中收到的有效断言		N/A
在有效 HTTP POST 中发送的有效断言		N/A
用户无须再次输入凭据就能被应用程序所验证		N/A
发送至合作伙伴产品（RP）并经过验证的有效已签名断言		N/A
Web SSO 断言中属性语句的成功验证		N/A

JEC　　*P=“通过” 或 “是” F=失败 N/A=不可用功能

** **注意：**XS40 并非 Web 应用程序，因此自身不具备设置身份验证 cookie 的规则。然而，基于目标 Web 应用程序及其 cookie 设置方法，您能够配置 XS40 以设置适当的 cookie，以便用户无须再次验证。XS40 能从 FIM 发布的 SAML 响应中提取主题信息，并使用此信息设置符合本地安全策略的 cookie。这将允许用户在不用再次输入凭据的情况下通过远程 Web 应用程序的验证。

8. 已知问题

无。